

WHAT IS CLAIMED IS:

5

1. A method of controlling a processor that changes an execution sequence of instructions arranged in a program, the method comprising the steps of:

10

executing a second instruction that is placed after a first instruction in the program, prior to execution of the first instruction; and

15

when an address of first data to be executed by the first instruction is included in an address region of second data to be processed by the second instruction, overwriting an execution result of the first instruction on data corresponding to the address of the first data.

20

2. The method as claimed in claim 1,

wherein:

25

the first instruction is a store instruction to store the first data into a storage unit; and

30

the second instruction is a load instruction to read out the second data from the storage unit.

35

3. The method as claimed in claim 1,

wherein:

the step of executing the second

COPIES 12500

instruction includes the step of storing information  
for specifying a storage unit that stores an address  
of the second data to be processed by the second  
instruction and a result obtained by the execution  
5 of the second instruction; and

the step of overwriting is carried out in  
accordance with the address of the data to be  
processed by the second instruction and the  
information for specifying the storage unit.

10

4. The method as claimed in claim 3,  
15 further comprising the step of executing a third  
instruction so as to erase the address and the  
information for specifying the storage unit.

20

5. The method as claimed in claim 3,  
further comprising the step of executing a third  
instruction so as to erase either the address of the  
25 data to be processed by the second instruction or  
the information for specifying the storage unit.

30

6. The method as claimed in claim 1,  
wherein:

the step of executing the second  
instruction includes the step of storing  
35 identification information of a context to be  
processed by the second instruction; and  
the step of overwriting is carried out in

accordance with the identification information of the context.

5

7. The method as claimed in claim 1, wherein the step of overwriting is carried out in accordance with an interrupt operation program, when  
10 the address of the first data to be processed by the first instruction is included in the address region of the second data to be processed by the second instruction.

15

8. The method as claimed in claim 1, wherein the step of overwriting is carried out in accordance with a program at a branch destination  
20 designated by executing a branch instruction, when the address of the first data to be processed by the first instruction is included in the address region of the second data to be processed by the second  
25 instruction.

30

9. A processor that executes programmed instructions, comprising:

a storage destination memory unit that stores a storage designation of a result obtained by executing a second instruction prior to execution of  
35 a first instruction, the second instruction being placed after the first instruction in a program;  
a judgment unit that determines whether or

not an address of first data to be processed by the first instruction is included in an address region of second data to be processed by the second instruction; and

- 5           a data restoration unit that overwrites a result obtained by executing the first instruction on the second data corresponding to the address of the first data at the storage destination stored in the storage destination memory unit, when the
- 10       judgment unit determines that the address of the first data is included in the address region of the second data.

15

10. The processor as claimed in claim 9, further comprising a storage unit that stores data, wherein:

- 20           the first instruction is a store instruction to store the first data into the storage unit; and
- the second instruction is a load instruction to read out the second data from the
- 25       storage unit.

- 30           11. The processor as claimed in claim 9, further comprising a plurality of storage units, the storage destination memory unit stores the information for specifying one of the storage units in which an address of the second data and the
- 35       result obtained by executing the second instruction is stored.

09736357-124500

12. The processor as claimed in claim 9, further comprising a context information storage information for specifying a context to be processed by the second instruction,

5 wherein the judgment unit is activated, only when a context to be processed by the first instruction is determined to coincide with the context to be processed by the second instruction, in accordance with the information stored in the  
10 context information storage unit.

13. The processor as claimed in claim 9,  
15 wherein the data restoration unit performs an overwrite operation in accordance with an interrupt operation program, when the judgment unit determines that the address of the first data is included in  
20 the address region of the second data.

14. The processor as claimed in claim 9,  
25 wherein the data restoration unit performs an overwrite operation in accordance with a program at a branch destination designated through execution of a branch instruction, when the judgment unit  
30 determines that the address of the first data is included in the address region of the second data.

35 15. The processor as claimed in claim 9, further comprising a storage destination erase unit

00735357-12150

that executes a third instruction so as to erase the storage destination stored in the storage destination memory unit.

5

16. The processor as claimed in claim 9,  
further comprising a storage destination erase unit  
10 that executes a third instruction so as to erase the storage destination stored in the storage destination memory unit.

15

17. A method of controlling a processor  
that controls execution of programmed instructions  
arranged in a program, the method comprising the  
20 steps of:

executing an instruction prior to  
execution of a branch instruction, the instruction  
being placed after the branch instruction in the  
program;

25 retaining an exception operation when  
necessity of the exception operation is detected in  
the step of executing;

performing the exception operation when  
the retained exception operation is needed in  
30 execution of an instruction at a branch destination  
selected through the execution of the branch  
instruction; and

returning to the program so as to continue  
the execution of the instruction at the branch  
35 destination when the exception operation is finished.

09726357-121500

18. A method of controlling a processor that controls execution of instructions arranged in a program,

- the method comprising the steps of:
- 5       executing an instruction prior to execution of a branch instruction, the instruction being placed after the branch instruction in the program;
- 10       retaining an exception operation when an exception start instruction that requires the exception operation is detected in the step of executing;
- 15       performing the exception operation when the retained exception operation is required in execution of an instruction at a branch destination selected through the execution of the branch instruction; and
- 20       returning to the program so as to sequentially execute the instructions starting from the exception start instruction, when the exception operation is finished.

- 25       19. The method as claimed in claim 17, wherein the step of performing the exception operation is carried out by executing an interrupt operation program.

30

- 35       20. The method as claimed in claim 17, wherein:
- the step of retaining the exception operation includes the step of storing information

for performing the retained exception operation; and  
the step of performing the exception  
operation is carried out in accordance with the  
stored information.

5

21. The method as claimed in claim 17,

10 wherein:

the step of retaining the exception  
operation includes the step of allocating  
identification information to each set of data  
obtained as a result of a predetermined operation,  
15 the identification information indicating whether or  
not the corresponding set of data requires the  
exception operation; and

the step of performing the exception  
operation is carried out when a set of data that is  
20 determined to require the exception operation from  
the identification information is processed in the  
execution of the instruction at the branch  
destination selected through the execution of the  
branch instruction.

25

22. The method as claimed in claim 20,

30 further comprising the step of executing a  
predetermined instruction so as to nullify the  
information for performing the retained exception  
operation.

35

0073357-1160



23. The method as claimed in claim 21,  
further comprising the step of executing a  
predetermined instruction so as to nullify the  
identification information.

5

24. The method as claimed in claim 21,  
10 further comprising the step of executing a  
predetermined instruction so as to read out the  
identification information or to rewrite the  
identification information.

15

25. The method as claimed in claim 18,  
wherein the step of performing the exception  
20 operation is carried out by executing an interrupt  
operation program.

25

26. The method as claimed in claim 18,  
wherein:  
the step of retaining the exception  
operation includes the step of storing information  
30 for performing the retained exception operation; and  
the step of performing the exception  
operation is carried out in accordance with the  
stored information.

35

0025357-121500

27. The method as claimed in claim 18,  
wherein:

- the step of retaining the exception  
operation includes the step of allocating  
5 identification information to each set of data  
obtained as a result of a predetermined operation,  
the identification information indicating whether or  
not the corresponding set of data requires the  
exception operation; and  
10 the step of performing the exception  
operation is carried out when a set of data that is  
determined to require the exception operation from  
the identification information is processed in the  
execution of the instruction at the branch  
15 destination selected through the execution of the  
branch instruction.

- 20 28. The method as claimed in claim 26,  
further comprising the step of executing a  
predetermined instruction so as to nullify the  
information for performing the retained exception  
25 operation.

- 30 29. The method as claimed in claim 27,  
further comprising the step of executing a  
predetermined instruction so as to nullify the  
identification information.

30. The method as claimed in claim 27,  
further comprising the step of executing a  
predetermined instruction so as to read out the  
identification information or to rewrite the  
5 identification information.

10 31. A processor that executes  
instructions arranged in a program, the processor  
comprising:  
a control unit that controls an execution  
sequence so that an instruction placed after a  
15 branch instruction in the program is executed prior  
to execution of the branch instruction;  
an exception inhibiting unit that retains  
an exception operation when necessity of the  
exception operation is detected during the execution  
20 of the instruction placed after the branch  
instruction;  
an exception operation unit that performs  
the exception operation when the exception operation  
retained by the exception inhibiting unit is needed  
25 in execution of an instruction at a branch  
destination selected through execution of the branch  
instruction; and  
a return unit that returns to the program  
when the exception operation is finished, and  
30 continues the execution of the instruction at the  
branch destination.

35 32. A processor that executes  
instructions arranged in a program, the processor

comprising:

- 5 a control unit that controls an execution sequence so that an instruction placed after a branch instruction in the program is executed prior to execution of the branch instruction;
- 10 an exception inhibiting unit that retains an exception operation when an exception start instruction that requires the exception operation is detected during the execution of the instruction placed after the branch instruction;
- 15 an exception operation unit that performs the exception operation when the exception operation retained by the exception inhibiting unit is needed in execution of an instruction at a branch destination selected through execution of the branch instruction; and
- 20 a return unit that returns to the program when the exception operation is finished, and sequentially executes the instructions starting from the exception start instruction.

- 25 33. The processor as claimed in claim 31, wherein the exception operation unit executes an interrupt operation program so as to perform the exception operation.

30

34. The processor as claimed in claim 31, further comprising a storage unit that stores information for performing the exception operation retained by the exception inhibiting unit,
- 35 wherein the exception operation unit

performs the exception operation in accordance with the information stored in the storage unit.

5

35. The processor as claimed in claim 31,

wherein:

- 10 the exception inhibiting unit allocates identification information to each set of data obtained as a result of a predetermined operation, the identification information indicating whether or not the exception operation is required; and
- 15 the exception operation unit performs the exception operation, when data determined to require the exception operation in accordance with the identification information is processed in the execution of the instruction at the branch destination selected through the execution of the
- 20 branch instruction.

- 25 36. The processor as claimed in claim 34, further comprising a history nullifying unit that executes a predetermined instruction so as to nullify the information for executing the retained exception operation.

30

- 35 37. The processor as claimed in claim 35, further comprising an identification information nullifying unit that executes a predetermined instruction so as to nullify the identification

00735357-14500

information.

5

38. The processor as claimed in claim 35,  
further comprising:

an identification information read unit  
that executes a predetermined instruction so as to  
10 read out the identification information; and  
an identification information rewrite unit  
that executes a predetermined instruction so as to  
rewrite the identification information.

15

39. The processor as claimed in claim 32,  
wherein the exception operation unit executes an  
20 interrupt operation program so as to perform the  
exception operation.

25

40. The processor as claimed in claim 32,  
further comprising a storage unit that stores  
information for performing the exception operation  
retained by the exception inhibiting unit,  
30 wherein the exception operation unit  
performs the exception operation in accordance with  
the information stored in the storage unit.

35

41. The processor as claimed in claim 32,

0072657-121500

wherein:

the exception inhibiting unit allocates identification information to each set of data obtained as a result of a predetermined operation, the identification information indicating whether or not the exception operation is required; and

the exception operation unit performs the

15

20

43. The processor as claimed in claim 41, further comprising an identification information nullifying unit that executes a predetermined instruction so as to nullify the identification information.

44. The processor as claimed in claim 41,  
further comprising:

an identification information read unit  
that executes a predetermined instruction so as to  
read out the identification information; and

- an identification information rewrite unit  
5 that executes a predetermined instruction so as to  
rewrite the identification information.

10

45. A method of controlling execution of  
instructions in a program, the method comprising the  
steps of:

- executing an instruction prior to  
15 execution of a branch instruction, the instruction  
being placed after the branch instruction in the  
program;

- retaining a break operation when necessity  
to suspend execution of the program is detected in  
20 the step of executing the instruction; and

- performing the break operation when the  
retained break operation is required in execution of  
an instruction at a branch destination selected  
through the execution of the branch instruction.

25

46. The method as claimed in claim 45,  
30 wherein:

the step of retaining a break operation  
includes the step of storing information for  
performing the retained break operation; and

- the step of performing the break operation  
35 is carried out in accordance with the stored  
information.

0073557-1-1000



47. The method as claimed in claim 46, further comprising the step of nullifying the stored information.

5

48. The method as claimed in claim 45, wherein:

10 the step of retaining a break operation includes the step of setting a predetermined value into a flag; and

15 the step of performing the break operation includes the step of referring to the value of the flag so as to determine whether or not the retained break operation is needed in execution of an instruction at a branch destination selected through the execution of the branch instruction.

20

49. The method as claimed in claim 48, further comprising the step of executing a  
25 predetermined instruction so as to nullify the value of the flag.

30

50. The method as claimed in claim 45, wherein the step of performing the break operation includes the step of executing the instruction at the branch instruction selected through the  
35 execution of the branch instruction, in accordance with an interrupt operation program.

0073357-121500

an exception inhibiting a break operation when necessity of suspending execution of the program is detected in execution of a predetermined instruction prior to execution of a branch instruction, the predetermined instruction being placed after the branch instruction in the program; and a control unit that performs the

program; and  
10        a break operation unit that performs the  
break operation when the break operation retained by  
the exception inhibiting unit is required in  
execution of an instruction at a branch destination  
selected through the execution of the branch  
15 instruction.

20           52. The processor as claimed in claim 51,  
further comprising a storage unit that stores  
information for performing the retained break  
operation,

operation, wherein the break operation unit performs  
25 the break operation in accordance with the  
information stored in the storage unit.

53. The processor as claimed in claim 52, further comprising a nullifying unit that nullifies the information stored in the storage unit.

wherein:

15

20

30